WHAT REALLY IS pWCET?
A RIGOROUS AXIOMATIC PROPOSAL

RTSS 2023
6 December 2023

Sergey Bozhko, Filip Marković, Georg von der Brüggen, and Björn Brandenburg
WHAT EXACTLY IS pWCET? AND HOW DOES IT RELATE TO pET?

Probabilistic Worst-Case Execution Time (pWCET)
Probabilistic Execution Time (pET)
We propose

➔ First **fully formal** definitions of pET and pWCET
➔ **Adequacy property** capturing the notion of "IID upper bound on pET"
➔ **Prove** that our proposal of pWCET is adequate in this sense
**THIS PAPER IN A NUTSHELL**

**We propose**
- First **fully formal** definitions of pET and pWCET
- **Adequacy property** capturing the notion of "IID upper bound on pET"
- **Prove** that our proposal of pWCET is adequate in this sense

**We formalized our proposal with the Coq proof assistant**
- Semantics of stochastic real-time systems
- Definitions of pET, pWCET, and the adequacy property
- **Machine-checked** proof that pWCET is adequate

What **exactly** is pWCET? And how does it relate to pET?

Probabilistic Worst-Case Execution Time (pWCET)

Probabilistic Execution Time (pET)
ANALYSIS OF REAL-TIME SYSTEMS: THE BIG PICTURE
What Really is pWCET? A Rigorous Axiomatic Proposal

THE BIG PICTURE

To get the predictions right, we need:

➔ Model with the right specification

➔ E.g., model must have WCETs to allow (classical) response-time analysis

➔ Correct model derivation

➔ Optimistic WCET bound

Wrong predictions

➔ Correct analysis

➔ Flawed analysis

Predictions
The Big Picture

To get the predictions right, we need:

➔ Model with the right **specification**

➔ *E.g.,* model must include WCETs to allow (classical) response-time analysis
THE BIG PICTURE

Worst-Case Execution Time (WCET)

To get the predictions right, we need:

➔ Model with the right **specification**
  ➔ *E.g.*, model must include WCETs to allow (classical) response-time analysis

➔ Correct model **derivation**
  ➔ Optimistic WCET bound \(\rightarrow\) Wrong predictions
THE BIG PICTURE

Worst-Case Execution Time (WCET)

To get the predictions right, we need:

➔ Model with the right specification
  ➔ E.g., model must include WCETs to allow (classical) response-time analysis

➔ Correct model derivation
  ➔ Optimistic WCET bound ➞ Wrong predictions

➔ Correct analysis
  ➔ Flawed analysis ➞ Wrong predictions

Model of the system

Derivation

Analysis

Predictions
To get the predictions right, we need:

➔ Model with the right **specification**
  ➔ *E.g.*, model must include WCETs to allow (classical) response-time analysis

➔ Correct model **derivation**
  ➔ Optimistic WCET bound ➔ Wrong predictions

➔ Correct **analysis**
  ➔ Flawed analysis ➔ Wrong predictions

Interpretation of common models is pretty straightforward in the deterministic case
SPECIFICATIONS ARE LESS OBVIOUS IN THE STOCHASTIC CASE

To get the predictions right, we need:

➔ Model with the right **specification**
  ➔ *E.g.*, model must include WCETs to allow (classical probabilistic) response-time analysis

➔ Correct model **derivation**
  ➔ Optimistic WCET bound ➔ Wrong predictions

➔ Correct **analysis**
  ➔ Flawed analysis ➔ Wrong predictions

\[ \mathbb{P} \ldots \]
To get the predictions right, we need:

➔ Model with the right **specification**
  ➔ *E.g.*, model must include WCETs to allow (classical probabilistic) response-time analysis

➔ Correct model **derivation**
  ➔ Optimistic WCET bound ➔ Wrong predictions

➔ Correct **analysis**
  ➔ Flawed analysis ➔ Wrong predictions

**SPECIFICATIONS ARE LESS OBVIOUS IN THE STOCHASTIC CASE**

Specifications of stochastic RTSs are much less straightforward
THE CASE FOR STOCHASTIC RTS

Stochastic real-time systems

→ Model of RTSs, where workload parameters are modelled stochastically
**THE CASE FOR STOCHASTIC RTS**

**Stochastic real-time systems**

→ Model of RTSs, where workload parameters are modelled *stochastically*

**Pros:**

→ Most systems *can tolerate deadline misses*  

   → Want to take advantage of this

Stochastic real-time systems

➔ Model of RTSs, where workload parameters are modelled *stochastically*

**Pros:**

➔ Most systems *can tolerate deadline misses*  
  ➔ Want to take advantage of this

➔ Allows answering *quantitative* questions

---

*THE CASE FOR STOCHASTIC RTS*

---

---

THE CASE FOR STOCHASTIC RTS

Stochastic real-time systems

→ Model of RTSs, where workload parameters are modelled stochastically

Pros:

→ Most systems can tolerate deadline misses
   → Want to take advantage of this

→ Allows answering quantitative questions

→ Enables analysis of transiently overloaded systems
   → Ubiquitous in practice
   → E.g., FMTV Challenge 2016

Question 14 For the most time-critical functions in the system, roughly how frequently can the deadline of a function be missed without causing a system failure. (n = 101)

The total utilization of that system goes above 100%. Using response time analysis in such situation automatically yields unbounded (infinite) worst-case response times.

DEPENDENCY IN STOCHASTIC RTS
THE PROBLEM OF STOCHASTIC RTS: DEPENDENCY
Probabilistic Execution Times (pETs)

Ground-truth behavior of jobs in the system

THE PROBLEM OF STOCHASTIC RTS: DEPENDENCY

Model of the system

Derivation

Analysis

pET

Predictions

Probabilistic Execution Times (pETs) are dependent.

Limits the application of probability theory tools. E.g., convolution is not applicable.

Model of the system

Derivation

Analysis

pET

Predictions

Ground-truth behavior of jobs in the system
Probabilistic Execution Times (pETs) are dependent!

Ground-truth behavior of jobs in the system

THE PROBLEM OF STOCHASTIC RTS: DEPENDENCY
Probabilistic Execution Times (pETs) are dependent!

- Image processing: Two consecutive frames might take similar amounts of compute

Ground-truth behavior of jobs in the system
**THE PROBLEM OF STOCHASTIC RTS: DEPENDENCY**

Probabilistic Execution Times (pETs) are **dependent**!

- Image processing: Two consecutive frames might take similar amounts of compute.
- Behavior of a prior job influences the state of the cache.

Ground-truth behavior of jobs in the system.

---

Model of the system → Derivation → Analysis

pET

Predictions
**THE PROBLEM OF STOCHASTIC RTS: DEPENDENCY**

**Probabilistic Execution Times (pETs) are dependent!**

- Image processing: Two consecutive frames might take similar amounts of compute
- Behavior of a prior job influences the state of the cache

Ground-truth behavior of jobs in the system

Q: Can we disregard dependency and continue anyway?

Model of the system

Derivation

Analysis

Predictions
Probabilistic Execution Times (pETs) are dependent!

→ Tia et al. 1995: computation times are not independent [1]
→ Ignoring this fact may lead to incorrect bounds

“Unfortunately, the computation times of individual requests are not statistically independent. […] As a consequence, the probability of meeting deadlines thus computed may be overly optimistic.”

Q: Can we disregard dependency and continue anyway?
A: No, the results can be optimistic

Probabilistic Execution Times (pETs) are dependent!

→ Tia et al. 1995: computation times are not independent \[1\]
  → Ignoring this fact may lead to incorrect bounds
  “Unfortunately, the computation times of individual requests are not statistically independent. [...] As a consequence, the probability of meeting deadlines thus computed may be overly optimistic.”

→ Limits the application of probability theory tools
  → E.g., convolution is not applicable

Q: Can we disregard dependency and continue anyway?
A: No, the results can be optimistic

PRIOR WORK: pWCET
PROBABILISTIC WORST-CASE EXECUTION TIME (pWCET)
We note that the actual execution times for a sequence of jobs of a task, which exercise the same or different paths, may well show strong correlations and dependences. It is the modelling of the execution times via an appropriate pWCET distribution which enables probabilistic independence to be assumed. (This is similar to the conventional case of a single WCET which can similarly be used in this way, even though the actual execution times of different jobs have strong dependences).

PROBABILISTIC WORST-CASE EXECUTION TIME (pWCET)

We note that the actual execution times for a sequence of jobs of a task, which exercise the same or different paths, may well show strong correlations and dependences. It is the modelling of the execution times via an appropriate pWCET distribution which enables probabilistic independence to be assumed. (This is similar to the conventional case of a single WCET which can similarly be used in this way, even though the actual execution times of different jobs have strong dependences).

pWCET is a convenient model abstraction to regain independence

Goal: enable "IID reasoning"

Independent and identically distributed

We note that the actual execution times for a sequence of jobs of a task, which exercise the same or different paths, may well show strong correlations and dependences. It is the modelling of the execution times via an appropriate pWCET distribution which enables probabilistic independence to be assumed. (This is similar to the conventional case of a single WCET which can similarly be used in this way, even though the actual execution times of different jobs have strong dependences).

PROBABILISTIC WORST-CASE EXECUTION TIME (pWCET)

pWCET is a **convenient** model abstraction to regain independence

- **Goal**: enable "IID reasoning"
- The **mainstream** approach to hiding dependence
- Unlocks powerful probability theory techniques
  - Such as convolution, Chernoff bound, etc.

We note that the actual execution times for a sequence of jobs of a task, which exercise the same or different paths, may well show **strong correlations and dependences**. It is the **modelling** of the execution times via an appropriate pWCET distribution which enables probabilistic independence to be assumed. (This is similar to the conventional case of a single WCET which can similarly be used in this way, even though the actual execution times of different jobs have strong dependences).

**PROBABILISTIC WORST-CASE EXECUTION TIME (pWCET)**

pWCET is a *convenient* model abstraction to regain independence

- **Goal**: enable "IID reasoning"
- The **mainstream** approach to hiding dependence
- Unlocks powerful probability theory techniques
  - Such as *convolution*, *Chernoff bound*, etc.
- .... but when **exactly** is a pWCET distribution "appropriate"?

We note that the actual execution times for a sequence of jobs of a task, which exercise the same or different paths, may well show *strong correlations and dependences*. It is the *modelling* of the execution times via an *appropriate* pWCET distribution which enables probabilistic independence to be assumed. (This is similar to the conventional case of a single WCET which can similarly be used in this way, even though the actual execution times of different jobs have strong dependences).

---

THE STATE-OF-THE-ART pWCET DEFINITION

Definition 2. The probabilistic Worst-Case Execution Time (pWCET) distribution for a task is the least upper bound, in the sense of the greater than or equal to operator $\geq$ (defined below), on the execution time distribution of the jobs of the task for every valid scenario of operation, where a scenario of operation is defined as an infinitely repeating sequence of input states (including both input values and software state variables) and initial hardware states that characterise a feasible way in which recurrent execution of the task may occur.

What Really is pWCET? A Rigorous Axiomatic Proposal

The STATE-OF-THE-ART pWCET DEFINITION

Definition 2. The probabilistic Worst-Case Execution Time (pWCET) distribution for a task is the least upper bound, in the sense of the greater than or equal to operator $\geq$ (see below), on the execution time distribution of the jobs of the task for every valid scenario of operation, where a scenario of operation is defined as an infinitely repeating sequence of input states (including both input values and software state variables) and initial hardware states that characterise a feasible way in which recurrent execution of the task may occur.

Side note: dominance relation $\leq$

$\rightarrow$ Proposed by Diaz et al. in 2004 [2]

$\rightarrow$ Partial order on random variables

$\rightarrow$ Similar to stochastic dominance

$\rightarrow$ $\mathcal{A} \leq \mathcal{B} := \forall x, \Pr[\mathcal{A} \leq x] \geq \Pr[\mathcal{B} \leq x]$

THE STATE-OF-THE-ART pWCET DEFINITION

Definition 2. The probabilistic Worst-Case Execution Time (pWCET) distribution for a task is the least upper bound, in the sense of the greater than or equal to operator $\succeq$ (defined below), on the execution time distribution of the jobs of the task for every valid scenario of operation, where a scenario of operation is defined as an infinitely repeating sequence of input states (including both input values and software state variables) and initial hardware states that characterise a feasible way in which recurrent execution of the task may occur.

Pros

➔ Gives the right intuition

➔ Identifies that "scenario of operation" is the key notion

pWCET DEFINITION IS OPEN TO INTERPRETATION

Definition 2. The probabilistic Worst-Case Execution Time (pWCET) distribution for a task is the least upper bound, in the sense of the greater than or equal to operator ≥ (defined below), on the execution time distribution of the jobs of the task for every valid scenario of operation, where a scenario of operation is defined as an infinitely repeating sequence of input states (including both input values and software state variables) and initial hardware states that characterise a feasible way in which recurrent execution of the task may occur.

Pros

➔ Gives the right intuition
➔ Identifies that "scenario of operation" is the key notion

Cons

➔ Open to interpretation 🤔
➔ Key aspects are stated in prose only
➔ Not suitable for formal verification

What Really is pWCET? A Rigorous Axiomatic Proposal

pWCET DEFINITION IS OPEN TO INTERPRETATION

Definition 2. The \textit{probabilistic Worst-Case Execution Time (pWCET)} distribution for a task is the least upper bound, in the sense of the greater than or equal to operator $\geq$ (defined below), on the execution time distribution of the jobs of the task for every valid \textit{scenario of operation}, where a \textit{scenario} of operation is defined as an infinitely repeating sequence of input states (including both input values and software state variables) and initial hardware states that characterise a feasible way in which recurrent execution of the task may occur.

Pros

➔ Gives the right intuition

➔ Identifies that "scenario of operation" is the key notion

Cons

➔ Open to interpretation 🤔

➔ Key aspects are stated in \textit{prose} only

➔ \textbf{Not} suitable for formal verification

➔ \textbf{Does not} necessarily enable IID-based analyses

SOTA pWCET DOES NOT ENABLE IID ANALYSIS

 Already noted in [1]

**A toy system:** [1]

- Time-predictable hardware
- System has **four** states
- State **cycling through** its four possible values
- Small variability in each of the states
- Starts with random state

---

What Really is pWCET? A Rigorous Axiomatic Proposal

A toy system: [1]
- Time-predictable hardware
- System has four states
- State cycling through its four possible values
- Small variability in each of the states
- Starts with random state

Resulting pET distribution: [1]
\[
\begin{pmatrix}
10 \pm 2 & 20 \pm 2 & 30 \pm 2 & 40 \pm 2 \\
1/4 & 1/4 & 1/4 & 1/4
\end{pmatrix}
\]

Valid pWCET distribution: [1]
\[
\begin{pmatrix}
12 & 22 & 32 & 42 \\
1/4 & 1/4 & 1/4 & 1/4
\end{pmatrix}
\]
A toy system: [1]

- Time-predictable hardware
- System has four states
- State cycling through its four possible values
- Small variability in each of the states
- Starts with random state

Resulting pET distribution: [1]
\[
\begin{pmatrix}
10 \pm 2 & 20 \pm 2 & 30 \pm 2 & 40 \pm 2 \\
1/4 & 1/4 & 1/4 & 1/4
\end{pmatrix}
\]

Valid pWCET distribution: [1]
\[
\begin{pmatrix}
12 & 22 & 32 & 42 \\
1/4 & 1/4 & 1/4 & 1/4
\end{pmatrix}
\]

Except that ....

- Smallest workload of four consecutive jobs:
  \[ (10 - 2) + (20 - 2) + (30 - 2) + (40 - 2) = 92 \]

\[ \mathbb{P} \left[ \sum_{4} p_{ET} \geq 92 \right] = 1 \]

**A toy system:** [1]

- Time-predictable hardware
- System has four states
- State **cycling through** its four possible values
- Small variability in each of the states
- Starts with random state

**SOTA pWCET DOES NOT ENABLE IID ANALYSIS**

- Resulting pET distribution: [1]
  \[
  \begin{pmatrix}
  10 \pm 2 \\
  20 \pm 2 \\
  30 \pm 2 \\
  40 \pm 2 \\
  1/4 \\
  1/4 \\
  1/4 \\
  1/4 
  \end{pmatrix}
  \]

- Valid pWCET distribution: [1]
  \[
  \begin{pmatrix}
  12 \\
  22 \\
  32 \\
  42 \\
  1/4 \\
  1/4 \\
  1/4 \\
  1/4 
  \end{pmatrix}
  \]

**Except that ....**

- Smallest workload of four consecutive jobs:
  \[
  (10 - 2) + (20 - 2) + (30 - 2) + (40 - 2) = 92
  \]

- Sum of four pWCETs is insufficient:
  \[
  E.g., 12 + 12 + 12 + 12 = 48 \text{ has nonzero probability}
  \]

A toy system: [1]

- Time-predictable hardware
- System has four states
- State cycling through its four possible values
- Small variability in each of the states
- Starts with random state

Except that ....

- Smallest workload of four consecutive jobs:
  - \((10 - 2) + (20 - 2) + (30 - 2) + (40 - 2) = 92\)
- Sum of four pWCETs is insufficient:
  - E.g., \(12 + 12 + 12 + 12 = 48\) has nonzero probability

Resulting pET distribution: [1]
\[
\begin{pmatrix}
10 ± 2 & 20 ± 2 & 30 ± 2 & 40 ± 2 \\
1/4 & 1/4 & 1/4 & 1/4 \\
\end{pmatrix}
\]

Valid pWCET distribution: [1]
\[
\begin{pmatrix}
12 & 22 & 32 & 42 \\
1/4 & 1/4 & 1/4 & 1/4 \\
\end{pmatrix}
\]

Not "appropriate" for IID-based analysis

A toy system:[1]

⇒ Time-predictable hardware
⇒ System has **four** states
⇒ State **cycling through** its four possible values
⇒ Small variability in each of the states
⇒ Starts with random state

⇒ Resulting pET distribution: [1]

\[
\begin{pmatrix}
10 \pm 2 & 20 \pm 2 & 30 \pm 2 & 40 \pm 2 \\
1/4 & 1/4 & 1/4 & 1/4
\end{pmatrix}
\]

⇒ (10 − 2) + (20 − 2) + (30 − 2) + (40 − 2) = 92
⇒ Sum of four pWCETs is insufficient:
⇒ E.g., 12 + 12 + 12 + 12 = 48 has nonzero probability

So, what is "appropriate" pWCET?

---

OUR PROPOSAL:

AXIOMATIC $pWCET$
DESIGN GOALS
**Formal** definitions of pET and pWCET

- *Definitions that are mathematically formal and unambiguous*
**DESIGN GOALS**

**Formal** definitions of pET and pWCET

- **Definitions that are mathematically formal and unambiguous**

**Adequacy property**: pWCET enables IID analysis

- **Any sound analysis assuming IID costs must result in a valid estimation**
**DESIGN GOALS**

**Formal** definitions of pET and pWCET
- Definitions that are mathematically formal and unambiguous

**Adequacy property**: pWCET enables IID analysis
- Any sound analysis assuming IID costs must result in a valid estimation

Precise enough to be **mechanisable** in Coq
- pET, pWCET and adequacy property with its proof must be formalisable in Coq proof assistant
AXIOMATIC pWCET
Definition 2. The probabilistic Worst-Case Execution Time (pWCET) distribution for a task is the least upper bound, in the sense of the greater than or equal to operator \( \succeq \) (defined below), on the execution time distribution of the jobs of the task for every valid scenario of operation, where a scenario of operation is defined as an infinitely repeating sequence of input states (including both input values and software state variables) and initial hardware states that characterise a feasible way in which recurrent execution of the task may occur.

AXIOMATIC pWCET

Definition 2. The **probabilistic Worst-Case Execution Time (pWCET)** distribution for a task is the least upper bound, in the sense of the greater than or equal to operator $\succeq$ (defined below), on the execution time distribution of the jobs of the task for every valid scenario of operation, where a *scenario* of operation is defined as an infinitely repeating sequence of input states (including both input values and software state variables) and initial hardware states that characterise a feasible way in which recurrent execution of the task may occur.

AXIOMATIC pWCET

Def. 7 (Ψ).

Definition 2. The probabilistic Worst-Case Execution Time (pWCET) distribution for a task is the least upper bound, in the sense of the greater than or equal to operator $\geq$ (defined below), on the execution time distribution of the jobs of the task for every valid scenario of operation, where a scenario of operation is defined as an infinitely repeating sequence of input states (including both input values and software state variables) and initial hardware states that characterise a feasible way in which recurrent execution of the task may occur.

**Definition 2.** The *probabilistic Worst-Case Execution Time (pWCET)* distribution for a task is the least upper bound, in the sense of the greater than or equal to operator $\succeq$ (defined below), on the execution time distribution of the jobs of the task for every valid scenario of operation, where a *scenario* of operation is defined as an infinitely repeating sequence of input states (including both input values and software state variables) and initial hardware states that characterise a feasible way in which recurrent execution of the task may occur.

---

**Def. 7 ( solidarity ).** A monotonically increasing function $F_t : \mathbb{W} \rightarrow [0, 1]$ with $F_t(0) = 0$ and $\lim_{t \to \infty} F_t(t) = 1$ is an axiomatic pWCET for a task $\tau_i$ if, for every $J \in \tau_i$ and every fixed arrival sequence $\xi \in \Xi$, there exists a partition $\mathcal{G}$ (Def. 4) such that

**Def. 4 ( partition ).** A partition $\mathcal{G} \triangleq \{ S_i \}_i$ is any finite, or countably infinite, disjoint cover of all positive-probability elements of $\Omega$.

---

What Really is pWCET? A Rigorous Axiomatic Proposal

Definition 2. The probabilistic Worst-Case Execution Time (pWCET) distribution for a task is the least upper bound, in the sense of the greater than or equal to operator $\geq$ (defined below), on the execution time distribution of the jobs of the task for every valid scenario of operation, where a scenario of operation is defined as an infinitely repeating sequence of input states (including both input values and software state variables) and initial hardware states that characterise a feasible way in which recurrent execution of the task may occur.


**AXIOMATIC pWCET**

Def. 4 (♦). A partition $\mathcal{G} \triangleq \{S_i\}_i$ is any finite, or countably infinite, disjoint cover of all positive-probability elements of $\Omega$.

Def. 7 (♣). A monotonically increasing function $F_i : \mathbb{W} \rightarrow [0, 1]$ with $F_i(0) = 0$ and $\lim_{t \rightarrow \infty} F_i(t) = 1$ is an axiomatic pWCET for a task $\tau_j$ if, for every $J \in \tau_j$ and every fixed arrival sequence $\xi \in \Xi$, there exists a partition $\mathcal{G}$ (Def. 4) such that

2) $F_i$ $\mathcal{G}$-dominates $C_J$ w.r.t. $\xi$ (Def. 6).

Def. 6 (♣). Given a job $J \in \mathcal{I}$, a fixed arrival sequence $\xi$, and a partition $\mathcal{G}$, a function $F : \mathbb{W} \rightarrow [0, 1]$ $\mathcal{G}$-dominates $C_J$ iff

$$\forall S_i \in \mathcal{G} \text{ s.t. } P[S_i \land \xi] > 0: \; F[C_J | S_i \land \xi] \geq F.$$
**Definition 2.** The probabilistic Worst-Case Execution Time (pWCET) distribution for a task is the least upper bound, in the sense of the greater than or equal to operator $\succeq$ (defined below), on the execution time distribution of the jobs of the task for every valid scenario of operation, where a scenario of operation is defined as an infinitely repeating sequence of input states (including both input values and software state variables) and initial hardware states that characterise a feasible way in which recurrent execution of the task may occur.

---

**AXIOMATIC pWCET**

**Def. 4 (⋆).** A partition $\mathcal{S} \triangleq \{S_i\}_i$ is any finite, or countably infinite, disjoint cover of all positive-probability elements of $\Omega$.

**Def. 5 (⋆).** Given a job $J \in \mathbb{J}$, a fixed arrival sequence $\xi$, and a partition $\mathcal{S}$, job $J$’s pET is partition-independent w.r.t. $\mathcal{S}$ iff, for any set $G \subseteq \mathbb{J}$ with $J \notin G$ and any fixed cost vector $\bar{c}_*$:

$$\forall S_i \in \mathcal{S} \text{ s.th. } P[S_i \wedge \xi] > 0:
\begin{align*}
P[C_J = \bar{c}_J \wedge \forall J' \in G: C_{J'} = \bar{c}_{J'} | S_i \wedge \xi] &= P[C_J = \bar{c}_J | S_i \wedge \xi] \cdot P[\forall J' \in G: C_{J'} = \bar{c}_{J'} | S_i \wedge \xi].
\end{align*}$$

**Def. 6 (⋆).** Given a job $J \in \mathbb{J}$, a fixed arrival sequence $\xi$, and a partition $\mathcal{S}$, a function $F: \mathbb{W} \rightarrow [0,1]$ dominates $C_J$ iff

$$\forall S_i \in \mathcal{S} \text{ s.th. } P[S_i \wedge \xi] > 0: F[C_J | S_i \wedge \xi] \leq F.$$

---

Definition 2. The probabilistic Worst-Case Execution Time (pWCET) distribution for a task is the least upper bound, in the sense of the greater than or equal to operator $\geq$ (defined below), on the execution time distribution of the jobs of the task for every valid scenario of operation, where a scenario of operation is defined as an infinitely repeating sequence of input states (including both input values and software state variables) and initial hardware states that characterise a feasible way in which recurrent execution of the task may occur.

Def. 7. A monotonically increasing function $F_t: \mathbb{W} \rightarrow [0,1]$ with $F_t(0) = 0$ and $\lim_{t \rightarrow \infty} F_t(t) = 1$ is an axiomatic pWCET for a task $\tau_i$, if, for every $J \in \tau_i$ and every fixed arrival sequence $\xi \in \Xi$, there exists a partition $\mathcal{G}$ (Def. 4) such that

1. $C_J$ is partition-independent w.r.t. $\xi$ and $\mathcal{G}$ (Def. 5), and
2. $F_t$ $\mathcal{G}$-dominates $C_J$ w.r.t. $\xi$ (Def. 6).

Def. 4. A partition $\mathcal{G} \triangleq \{S_i\}_i$ is any finite, or countably infinite, disjoint cover of all positive-probability elements of $\Omega$.

Def. 5. Given a job $J \in \mathbb{J}$, a fixed arrival sequence $\xi$, and a partition $\mathcal{G}$, job $J$’s pET is partition-independent w.r.t. $\mathcal{G}$, if for any set $G \subseteq \mathbb{J}$ with $J \notin G$ and any fixed cost vector $c_\xi$:

$$\forall S_i \in \mathcal{G} \text{ s.th. } P[S_i \land \xi] > 0:$$

$$P[C_J = \epsilon_J \land \forall J^' \in G: C_{J^'} = \epsilon_{J^'} | S_i \land \xi]$$

$$= P[C_J = \epsilon_J | S_i \land \xi] \cdot P[\forall J^' \in G: C_{J^'} = \epsilon_{J^'} | S_i \land \xi].$$

Def. 6. Given a job $J \in \mathbb{J}$, a fixed arrival sequence $\xi$, and a partition $\mathcal{G}$, a function $F: \mathbb{W} \rightarrow [0,1]$ $\mathcal{G}$-dominates $C_J$ iff

$$\forall S_i \in \mathcal{G} \text{ s.th. } P[S_i \land \xi] > 0:$$

$$F[C_J | S_i \land \xi] \leq F.$$
**Definition 2.** The probabilistic Worst-Case Execution Time (pWCET) distribution for a task is the least upper bound, in the sense of the greater than or equal to operator $\geq$ (defined below), on the execution time distribution of the jobs of the task for every valid scenario of operation, where a *scenario* of operation is defined as an infinitely repeating sequence of input states (including both input values and software state variables) and initial hardware states that characterise a feasible way in which recurrent execution of the task may occur.

**AXIOMATIC pWCET**

**Def. 7 (†).** A monotonically increasing function $F_1: \mathbb{W} \to [0, 1]$ with $F_1(0) = 0$ and $\lim_{t \to \infty} F_1(t) = 1$ is an axiomatic pWCET for a task $\tau_i$ if, for every $J \in \tau_i$ and every fixed arrival sequence $\xi \in \Xi$, there exists a partition $\mathcal{G}$ (Def. 4) such that

1. $C_J$ is partition-independent w.r.t. $\xi$ and $\mathcal{G}$ (Def. 5), and
2. $F_1 \mathcal{G}$-dominates $C_J$ w.r.t. $\xi$ (Def. 6).

**Def. 5 (†).** Given a job $J \in \mathbb{J}$, a fixed arrival sequence $\xi$, and a partition $\mathcal{G}$, job $J$’s pET is partition-independent w.r.t. $\mathcal{G}$ iff, for any set $G \subseteq \mathbb{J}$ with $J \notin G$ and any fixed cost vector $\bar{c}_x$:

$$\forall S_l \in \mathcal{G} \text{ s.th. } \Pr[S_l \wedge \xi] > 0:
\Pr[C_J = \bar{c}_J \wedge \forall J' \in G: C_{J'} = \bar{c}_{J'} | S_l \wedge \xi] = \Pr[C_J = \bar{c}_J | S_l \wedge \xi] \cdot \Pr[\forall J' \in G: C_{J'} = \bar{c}_{J'} | S_l \wedge \xi].$$

**Def. 4 (†).** A partition $\mathcal{G} \triangleq \{S_l\}_l$ is any finite, or countably infinite, disjoint cover of all positive-probability elements of $\Omega$.

**Def. 6 (†).** Given a job $J \in \mathbb{J}$, a fixed arrival sequence $\xi$, and a partition $\mathcal{G}$, a function $F: \mathbb{W} \to [0, 1]$ $\mathcal{G}$-dominates $C_J$ iff

$$\forall S_l \in \mathcal{G} \text{ s.th. } \Pr[S_l \wedge \xi] > 0: F(C_J | S_l \wedge \xi) \leq F.$$
ADEQUACY
ADEQUACY: FORMAL BASIS FOR IID REASONING

How do we know that an IID-based analysis that uses axiomatic pWCET will obtain a sound bound?
Intuitively, we want to prove:

\(\rightarrow\) Ground-truth pRT is \(\leq\)-bounded

by pRT derived via pWCETs

ADEQUACY: FORMAL BASIS FOR IID REASONING

How do we know that an IID-based analysis that uses axiomatic pWCET will obtain a sound bound?
ADEQUACY: FORMAL BASIS FOR IID REASONING

How do we know that an IID-based analysis that uses axiomatic pWCET will obtain a sound bound?

Intuitively, we want to prove:

→ Ground-truth pRT is \( \leq \)-bounded by pRT derived via pWCETs
ADEQUACY: FORMAL BASIS FOR IID REASONING

How do we know that an IID-based analysis that uses axiomatic pWCET will obtain a sound bound?

Intuitively, we want to prove:

$\Rightarrow$ Ground-truth pRT is $\leq$-bounded by pRT derived via pWCETs

pRT of $J_{i,j}$ obtained by any valid IID-based analysis using axiomatic pWCET
ADEQUACY: FORMAL BASIS FOR IID REASONING

How do we know that an IID-based analysis that uses axiomatic pWCET will obtain a sound bound?

Intuitively, we want to prove:

→ Ground-truth pRT is \( \leq \)-bounded by pRT derived via pWCETs

pRT of \( J_{i,j} \) obtained by any valid IID-based analysis using axiomatic pWCET
How do we know that an IID-based analysis that uses axiomatic pWCET will obtain a sound bound?

Intuitively, we want to prove:

\[ \Rightarrow \text{Ground-truth pRT is } \leq \text{-bounded by pRT derived via pWCETs} \]

Formal statement is **surprisingly** tricky and involves the notion of "replacement" of pETs with pWCETs.

\[ \forall J_{i,j}: R_{i,j} \preceq R_{i,j}^* \]

\[ \text{pRT of } J_{i,j} \text{ obtained by any valid IID-based analysis using axiomatic pWCET} \]
AXIOMATIC pWCET IS ADEQUATE

**Theorem (paraphrased).** Consider a job $J_{i,j}$. Let $\mathcal{R}_{i,j}$ be the pRT of $J_{i,j}$ in the initial system and $\mathcal{R}^*_{i,j}$ be the pRT of $J_{i,j}$ in a simplified system obtained via pWCET $F_i$. Then $\mathcal{R}_{i,j} \preceq \mathcal{R}^*_{i,j}$. 
AXIOMATIC pWCET IS ADEQUATE

**Theorem (paraphrased).** Consider a job $J_{i,j}$. Let $\mathcal{R}_{i,j}$ be the pRT of $J_{i,j}$ in the initial system and $\mathcal{R}_{i,j}^\star$ be the pRT of $J_{i,j}$ in a simplified system obtained via pWCET $F_i$. Then $\mathcal{R}_{i,j} \leq \mathcal{R}_{i,j}^\star$.

**Hint:**
1. Use axiomatic pWCET to construct a "copy" of the initial system, where pETs are replaced with job costs that are, by construction, IID and have distribution $F_i$.
2. Prove that pRT $\mathcal{R}_{i,j}^\star$ in the simplified system stochastically dominates the original pRT $\mathcal{R}_{i,j}$.
AXIOMATIC pWCET IS ADEQUATE

Theorem (paraphrased). Consider a job $J_{i,j}$. Let $R_{i,j}$ be the pRT of $J_{i,j}$ in the initial system and $R_{i,j}^*$ be the pRT of $J_{i,j}$ in a simplified system obtained via pWCET $F_i$. Then $R_{i,j} \leq R_{i,j}^*$.

Hint:
1. Use axiomatic pWCET to construct a "copy" of the initial system, where pETs are replaced with job costs that are, by construction, IID and have distribution $F_i$.
2. Prove that pRT $R_{i,j}^*$ in the simplified system stochastically dominates the original pRT $R_{i,j}$. 
AXIOMATIC pWCET IS ADEQUATE

**Theorem (paraphrased).** Consider a job $J_{i,j}$. Let $\mathcal{R}_{i,j}$ be the pRT of $J_{i,j}$ in the initial system and $\mathcal{R}_{i,j}^*$ be the pRT of $J_{i,j}$ in a simplified system obtained via pWCET $F_i$. Then $\mathcal{R}_{i,j} \leq \mathcal{R}_{i,j}^*$.

**Hint:**
1. Use axiomatic pWCET to construct a "copy" of the initial system, where pETs are replaced with job costs that are, by construction, IID and have distribution $F_i$.
2. Prove that pRT $\mathcal{R}_{i,j}^*$ in the simplified system stochastically dominates the original pRT $\mathcal{R}_{i,j}$.
Theorem (paraphrased). Consider a job $J_{i,j}$. Let $\mathcal{R}_{i,j}$ be the pRT of $J_{i,j}$ in the initial system and $\mathcal{R}_{i,j}^*$ be the pRT of $J_{i,j}$ in a simplified system obtained via pWCET $F_i$. Then $\mathcal{R}_{i,j} \leq \mathcal{R}_{i,j}^*$.

Hint:
1. Use axiomatic pWCET to construct a "copy" of the initial system, where pETs are replaced with job costs that are, by construction, IID and have distribution $F_i$.
2. Prove that pRT $\mathcal{R}_{i,j}^*$ in the simplified system stochastically dominates the original pRT $\mathcal{R}_{i,j}$. 

AXIOMATIC pWCET IS ADEQUATE
AXIOMATIC pWCET IS ADEQUATE

**Theorem (paraphrased).** Consider a job $J_{i,j}$. Let $R_{i,j}$ be the pRT of $J_{i,j}$ in the initial system and $R^*_{i,j}$ be the pRT of $J_{i,j}$ in a simplified system obtained via pWCET $F_i$. Then $R_{i,j} \leq R^*_{i,j}$.

**Hint:**
1. Use axiomatic pWCET to construct a "copy" of the initial system, where pETs are replaced with job costs that are, by construction, IID and have distribution $F_i$.
2. Prove that pRT $R^*_{i,j}$ in the simplified system stochastically dominates the original pRT $R_{i,j}$.
AXIOMATIC pWCET IS ADEQUATE

**Theorem** *(paraphrased)*. Consider a job $J_{i,j}$. Let $R_{i,j}$ be the pRT of $J_{i,j}$ in the initial system and $R_{i,j}^*$ be the pRT of $J_{i,j}$ in a simplified system obtained via pWCET $F_i$. Then $R_{i,j} \leq R_{i,j}^*$.

**Hint:**
1. Use axiomatic pWCET to construct a "copy" of the initial system, where pETs are replaced with job costs that are, by construction, IID and have distribution $F_i$.
2. Prove that pRT $R_{i,j}^*$ in the simplified system stochastically dominates the original pRT $R_{i,j}$.
AXIOMATIC pWCET IS ADEQUATE

**Theorem (paraphrased).** Consider a job $J_{i,j}$. Let $R_{i,j}$ be the pRT of $J_{i,j}$ in the initial system and $R_{i,j}^*$ be the pRT of $J_{i,j}$ in a simplified system obtained via pWCET $F_i$. Then $R_{i,j} \leq R_{i,j}^*$.

**Hint:**
1. Use axiomatic pWCET to construct a "copy" of the initial system, where pETs are replaced with job costs that are, by construction, IID and have distribution $F_i$.
2. Prove that pRT $R_{i,j}^*$ in the simplified system stochastically dominates the original pRT $R_{i,j}$.
**Theorem (paraphrased).** Consider a job $J_{i,j}$. Let $R_{i,j}$ be the pRT of $J_{i,j}$ in the initial system and $R_{i,j}^*$ be the pRT of $J_{i,j}$ in a simplified system obtained via pWCET $F_i$. Then $R_{i,j} \leq R_{i,j}^*$.

**Hint:**
1. Use axiomatic pWCET to construct a "copy" of the initial system, where pETs are replaced with job costs that are...
The LHS and RHS of the inequality can be simplified to \( \mathbb{P}[\mathcal{C}_{j_0} > c_0|\xi \land S_i] \) and \( \mathbb{P}_f[\mathcal{C}_{j_0} > c_0] \), respectively. Using the fact that \( \mathbb{P}[a > b] \leq \mathbb{P}[c > d] \iff \mathbb{P}[a \leq b] \geq \mathbb{P}[c \leq d] \), we transform the inequality to obtain (\( \bigstar \)):

\[ \mathbb{P}[\mathcal{C}_{j_0} \leq c_0|\xi \land S_i] \geq \mathbb{P}_f[\mathcal{C}_{j_0} \leq c_0]. \]

Finally, by construction (Def. 10), \( \mathbb{P}_f[\mathcal{C}_{j_0} \leq c_0] = F_i(c_0) \). Hence, we end up with \( \mathbb{P}[\mathcal{C}_{j_0} \leq c_0|\xi \land S_i] \geq F_i(c_0) \), which follows (\( \bigstar \)) from partition-dominance (Def. 6).

**Step 13**

In the last step, we exploit the top-level assumption \( H_{pWCET\_bounds\_cond\_cdf} \) to finish the proof.

**Section Step13.**

Notice that the following statement is very close to the pWCET guarantee \( H_{pWCET\_bounds\_cond\_cdf} \).

```
Lemma transformation_is_pRT_monotone_step13 :
  P<\mu_of S;[[ e j_rep (<=) c0 | \xi f n Sf ]] \geq
  P<\mu_{tsk}>{{ e \_pWCET (<=) c0 }}.
```

Also, note that we did not make any new assumptions in this section; hence, we are done.

**End Step13.**

**Clickable links to Coq specification**

→ Each definition, lemma, and proof step is accompanied by a link to the corresponding Coq specification

Links are cumbersome and not clickable in the official IEEE version

Links are directly clickable in the version provided on the author websites
Clickable links to Coq specification

→ Each definition, lemma, and proof step is accompanied by a link to the corresponding Coq specification

→ Links are cumbersome and not clickable in the official IEEE version

→ Links are directly clickable in the version provided on the author websites

The LHS and RHS of the inequality can be simplified to $P[\chi_{J_o} > c_0] \land S_i$ and $P_f[\tilde{\chi}_{J_o} > c_0]$, respectively. Using the fact that $P[a \land b] \leq P[c \land d] \iff P[a] \land b \geq P[c] \land d$, we transform the inequality to obtain ($\heartsuit$):

$$P(\chi_{J_o} \leq c_0) \land S_i \geq P_f(\tilde{\chi}_{J_o} \leq c_0).$$

Finally, by construction (Def. 10), $P_f[\tilde{\chi}_{J_o} \leq c_0] = F_i(c_0)$. Hence, we end up with $P[\chi_{J_o} \leq c_0] \land S_i \geq F_i(c_0)$, which follows ($\heartsuit$) from partition-dominance (Def. 6). □

---

Step 13

In the last step, we exploit the top-level assumption $H_{pWCET\_bounds\_cond\_cdf}$ to finish the proof.

Section Step13.

Notice that the following statement is very close to the pWCET guarantee $H_{pWCET\_bounds\_cond\_cdf}$.

Lemma transformation_is_pRT_monotone_step13 :

```
P<tt_of_S=(\{ e' \_rep (\ltself \_c0 \\_n Sf \}\}\_f = \_tsk=\{ e\_pWCET (\ltself \_c0 \}\}).
```

Also, note that we did not make any new assumptions in this section; hence, we are done.

End Step13.
CONCLUSION
CONCLUSION AND FUTURE WORK

What we did:

➔ First **fully formal** definitions of pET and pWCET
➔ **Adequacy property**: formalization of "safe IID upper bound on pET"
➔ **Prove** that our pWCET proposal is adequate
➔ All **mechanized** with Coq
CONCLUSION AND FUTURE WORK

What we did:

→ First **fully formal** definitions of pET and pWCET
→ **Adequacy property**: formalization of "safe IID upper bound on pET"
→ **Prove** that our pWCET proposal is adequate
→ All **mechanized** with Coq

Are there better definitions?

› *Maybe..?*
› *Please propose your preferred definition*
   
   ... and present an adequacy proof

---

**MPI-SWS**

*Sergey Bozhko, Filip Marković, Georg von der Brüggen, and Björn Brandenburg*
CONCLUSION AND FUTURE WORK

What we did:
→ First **fully formal** definitions of pET and pWCET
→ **Adequacy property**: formalization of "safe IID upper bound on pET"
→ **Prove** that our pWCET proposal is adequate
→ All **mechanized** with Coq

Are there **better** definitions?
› Maybe..?
› **Please propose your preferred definition**
   ... and present an adequacy proof

How to **derive** such axiomatic pWCET?
› Are existing methods compatible with it?
   (MBPTA? EVT? SPTA?)
› Can compatibility be proven in Coq?
What Really is pWCET? A Rigorous Axiomatic Proposal

CONCLUSION AND FUTURE WORK

What we did:

→ First **fully formal** definitions of pET and pWCET
→ **Adequacy property**: formalization of "safe IID upper bound on pET"
→ **Prove** that our pWCET proposal is adequate
→ All **mechanized** with Coq

Are there better definitions?

› **Maybe..?**
› **Please propose your preferred definition**
  ... and present an adequacy proof

How to derive such axiomatic pWCET?

› Are existing methods compatible with it? (MBPTA? EVT? SPTA?)
› Can compatibility be proven in Coq?

Use axiomatic pWCET to build an **analysis**

› **We prove that any sound analysis results in valid bounds. Let's verify one!**
What Really is pWCET? A Rigorous Axiomatic Proposal

CONCLUSION AND FUTURE WORK

What we did:

→ First **fully formal** definitions of pET and pWCET
→ **Adequacy property**: formalization of "safe IID upper bound on pET"
→ **Prove** that our pWCET proposal is adequate
→ All **mechanized** with Coq

Still do not like pWCET?

› **Come watch Filip's talk about pWCET-less Correlation-Tolerant Analysis** on December 8th (Session 11 @ 12:35pm)

Are there **better** definitions?

› Maybe..?
› Please propose your preferred definition
   ... and present an adequacy proof

How to **derive** such axiomatic pWCET?

› Are existing methods compatible with it?
   (MBPTA? EVT? SPTA?)
› Can compatibility be proven in Coq?

Use axiomatic pWCET to build an **analysis**

› We prove that any sound analysis results in valid bounds. Let's verify one!

---

Sergey Bozhko, Filip Marković, Georg von der Brüggen, and Björn Brandenburg

coq.inria.fr
BACKUP SLIDES
Theorem (paraphrased). Consider a job $J_{i,j}$. Let $R_{i,j}$ be the pRT of $J_{i,j}$ in the initial system and $R_{i,j}^*$ be the pRT of $J_{i,j}$ in a simplified system obtained via pWCET $F_i$. Then $R_{i,j} \leq R_{i,j}^*$.

Hint:
1. Use axiomatic pWCET to construct a "copy" of the initial system, where pETs are replaced with job costs that are, by construction, IID and have distribution $F_i$.
2. Prove that pRT $R_{i,j}^*$ in the simplified system stochastically dominates the original pRT $R_{i,j}$.

Def. 7 (**). A monotonically increasing function $F_i : \mathbb{W} \rightarrow [0,1]$ with $F_i(0) = 0$ and $\lim_{t \to \infty} F_i(t) = 1$ is an axiomatic pWCET for a task $\tau_i$ if, for every $J \in \tau_i$ and every fixed arrival sequence $\xi \in \Xi$, there exists a partition $\mathcal{G}$ (Def. 4) such that
1. $C_J$ is partition-independent w.r.t. $\xi$ and $\mathcal{G}$ (Def. 5), and
2. $F_i \mathcal{G}$-dominates $C_J$ w.r.t. $\xi$ (Def. 6).
TWO TYPES OF pWCET

**Dominance pWCET** [1]

→ \( F_i : \mathbb{W} \rightarrow [0,1] \)

→ Given \( c \), \( F_i(c) \) defines a bound on probability of a job of task \( \tau_i \) to have cost exceeding \( c \)

If \( F_i(50) = 0.999 \), then out of 100,000 jobs, at most 100 jobs are expected to have cost greater than 50

**Confidence pWCET** [2]

→ \( F_i : \mathbb{W} \rightarrow [0,1] \)

→ Given \( c \), \( F_i(c) \) defines a bound on probability that WCET of task \( \tau_i \) does not exceed \( c \)

If \( F_i(50) = 0.999 \), no job is expected to have cost greater than 50 and we are 99.9% confident about it